Developing Robust Digital Twins and Reinforcement Learning for Accelerator Control Systems at the Fermilab Booster

D. Kafkes *, C. Herwig, W. Pellico, G. Perdue, A. Quintero-Parra, B. Schupbach, K. Seiya, J. St. John, N. Tran (Fermi National Accelerator Laboratory, Batavia, IL USA); M. Schram (Thomas Jefferson National Accelerator Facility, Newport News, VA USA); Y. Huang (Pacific National Northwest Laboratory, Richland, WA USA); J. Duarte (University of California, San Diego, San Diego, CA USA); R. Keller (Columbia University, New York City, NY USA)

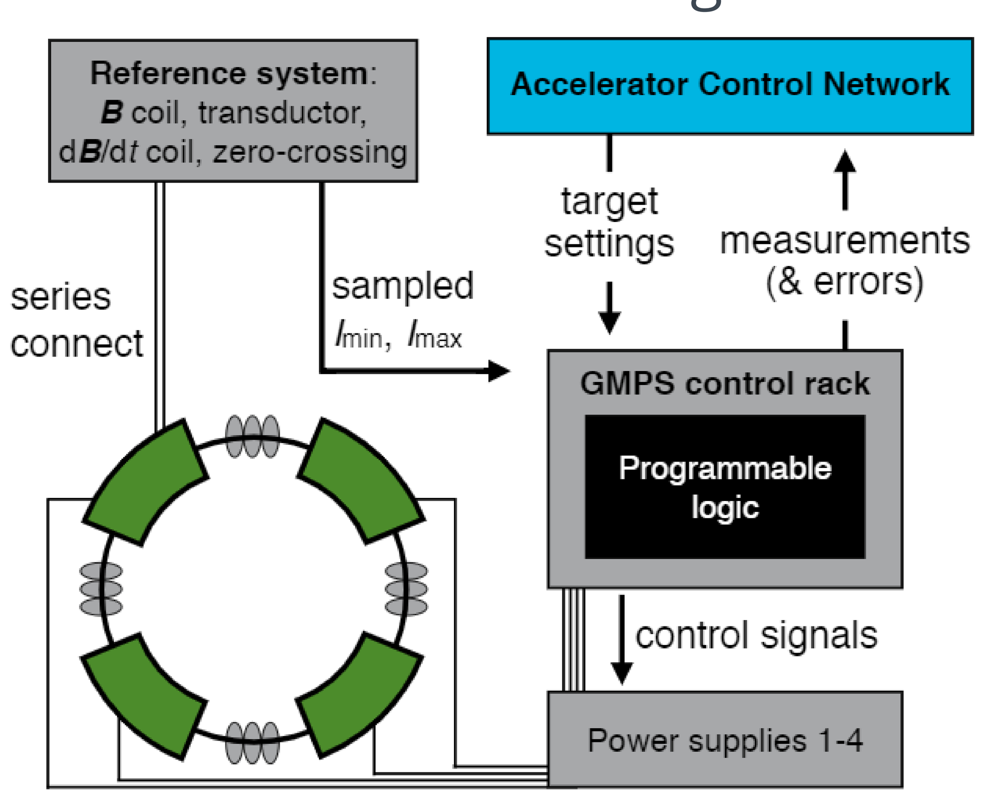
Project conducted under the auspices of FNAL-LDRD-2019-027 and DE-AC02-07CH11359.

ABSTRACT

We describe the offline machine learning development for study that aims to precisely regulate the Fermilab Booster Gradient Magnet Power Supply (GMPS) via a Field-Programmable Gate Array (FPGA). As part of this effort, we created a digital twin of the GMPS control system by training a Long Short-Term Memory (LSTM) to capture its full dynamics. We outline the path we took to carefully validate our digital twin before deploying it as a reinforcement learning (RL) environment. Additionally, we demonstrate the use of a Deep Q-Network (DQN) policy model with the capability to regulate the GMPS against realistic time-varying perturbations.

GOALS

Our intent is to develop an RL pipeline that can regulate the Booster Gradient Magnet Power Supply (GMPS)

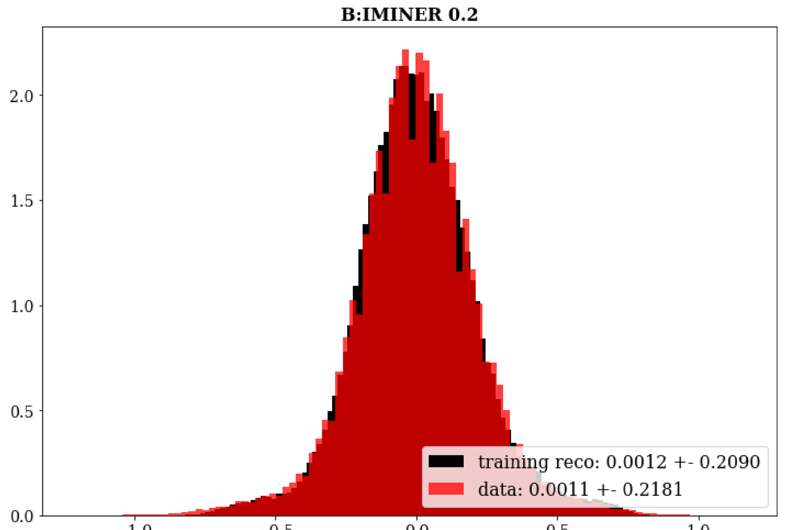


current better than the presently implemented PID controller **[1]**. The controller aims to regulate ***B:IMINER***, the deviation between the next 15Hz cycle minimum current reading (***B:VIMIN***) and its setting (***B_VIMIN***) using previous cycle values and the integral (γ) and proportional (α) gains **[2, 3]**.

$$\beta_t = \beta_{t-1} + \gamma_t \times \mathtt{B{:}IMINER}_{t-1}$$

$$\mathtt{B{:}VIMIN}_t = \mathtt{B\_VIMIN}_{t-1} - \alpha_t \times \mathtt{B{:}IMINER}_{t-1} - \beta_t$$

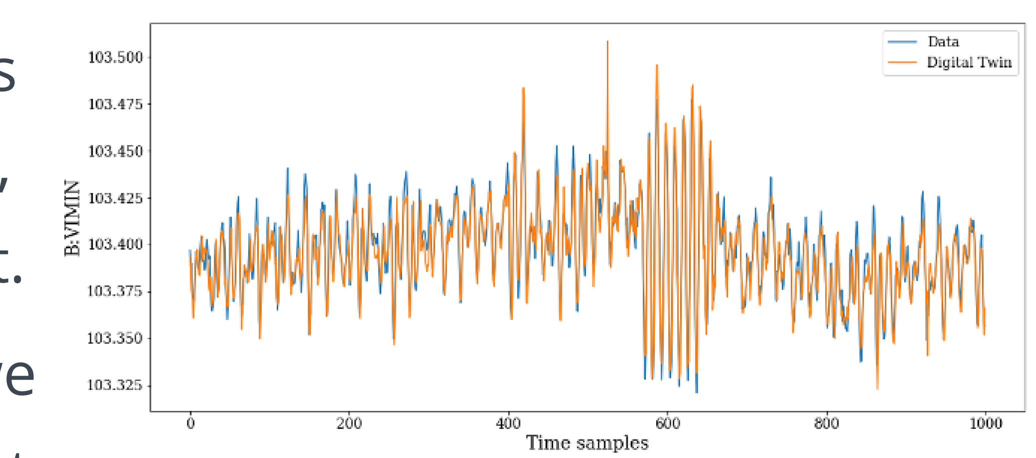
DIGITAL TWIN DEVELOPMENT AND UNCERTAINTY QUANTIFICATION



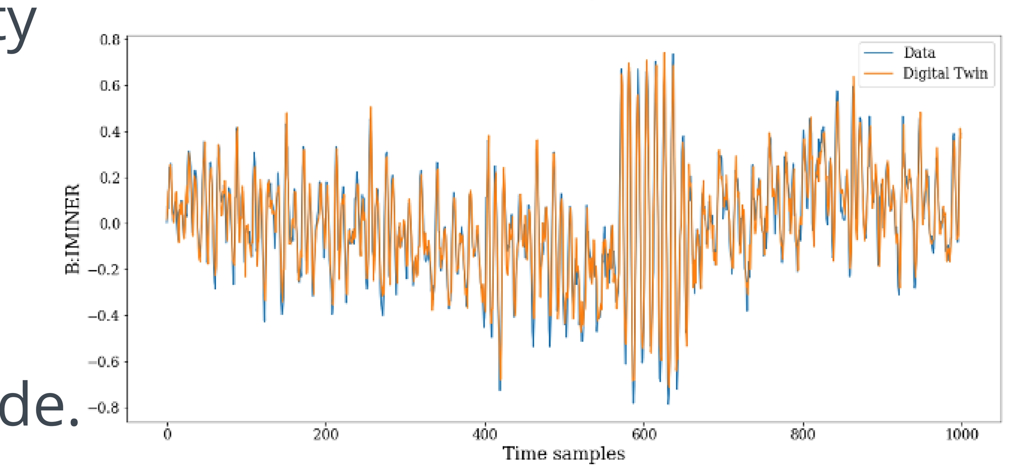
We first developed a stacked LSTM model to reproduce the behaviors of the GMPS system, thereby establishing an environment to train our RL algorithm **[1]**. We experimented with different time lookback windows, scalers, variable inclusion, and signal decomposition when crafting our model inputs and determined that a composed 6 to 2 model, including:

B:VIMIN + B:IMINER + B_VIMIN + B:LINFRQ + I:IB + I:MDAT40 ⟶ B:VIMIN + B:IMINER

with 1 second lookback and MinMax scaling performs best. Here ***B:LINFRQ*** is the 60Hz line frequency offset, and ***I:IB/ I:MDAT40*** measure the main injector current.

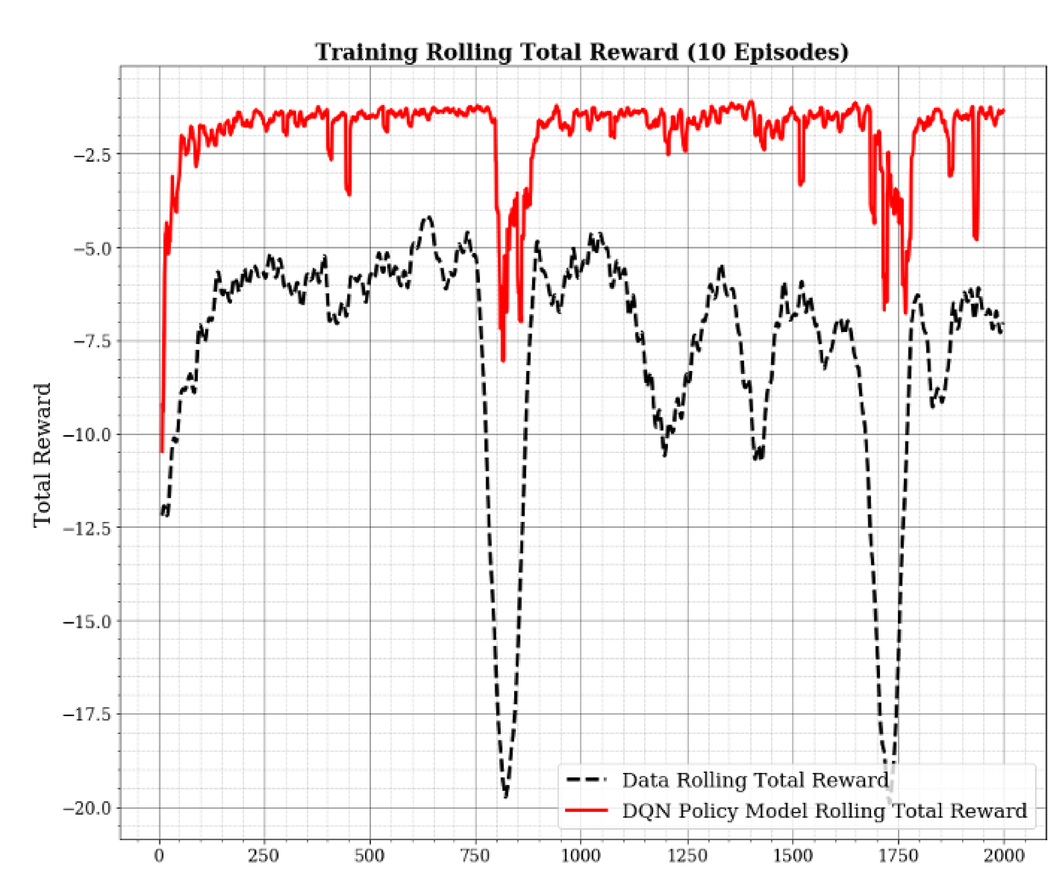


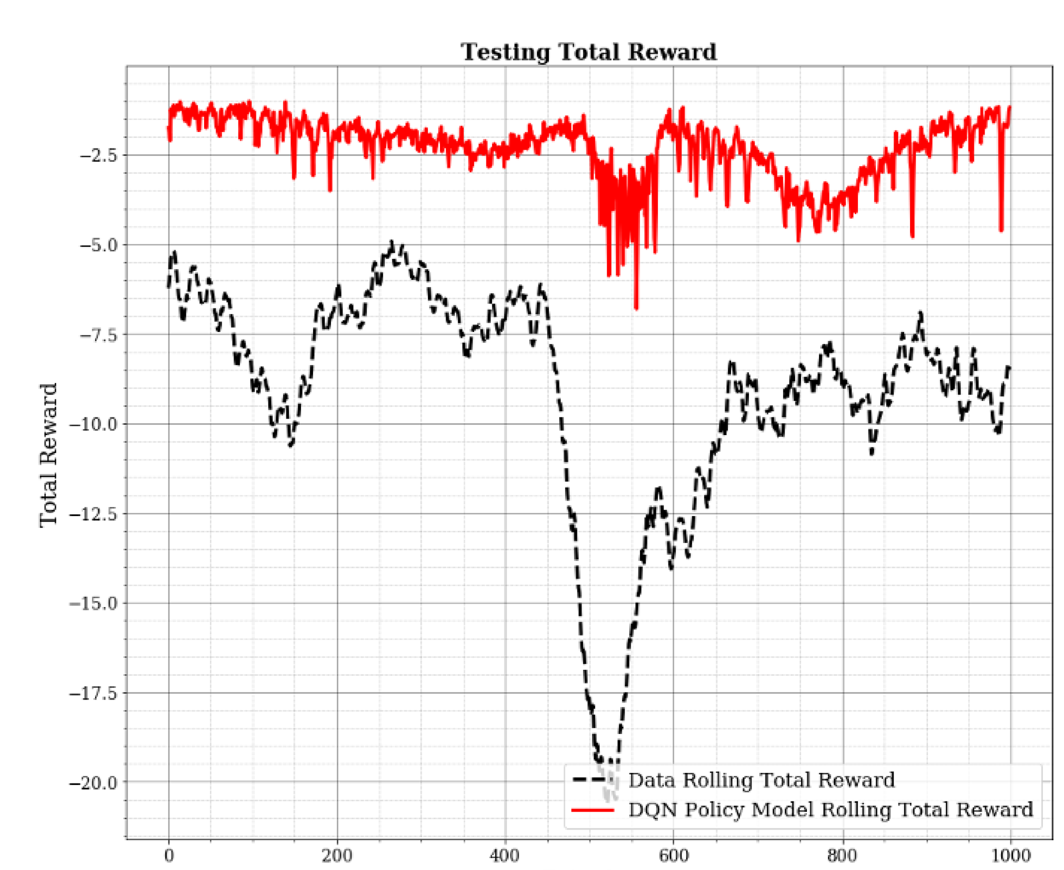
Additionally, after training our environment model, we performed concrete dropout as a means of uncertainty quantification **[4]**. We found that an intermediate dropout layer with probability .2 after the first LSTM layer gave us our best results in inference mode.



PRELIMINARY RL RESULTS

We present our most recent RL results, training a DQN as our policy model in our verified digital twin environment. The DQN approach involves training a deep neural network to learn the RL action-value function, which maps agent actions to rewards, and is usually deployed in environments that take discrete control actions **[5]**. We define our reward as -|***B:IMINER***| **[1]**. When comparing the DQN results to the PID controller, we see substantial improvement.





CONCLUSION

We outlined the steps we took to carefully validate our digital twin— perhaps the most important aspect of our machine learning development. Without a robust surrogate model to support offline training, we would not be able to trust deploying the trained agent on the live system in the future.

Citations **[1]** J. St. John, et al., *Real-time Artificial Intelligence for Accelerator Control* (2020), **[2]** J. Crawford, et al., *Booster Rookie Manual v4.1* (2009), **[3]** R. Keller, *Controlling Currents Report* (2019), **[4]** Y. Gal, J. Hron, & A. Kendall, *Concrete Dropout* (2017), **[5]** V. Mnih, et al., *Human-level control through deep reinforcement learning* (2015).

FERMILAB-POSTER-21-009-AD-SCD